

〈通用即时到账接口集成教程——PHP 版本〉

| | |
|-------|-------------|
| 文件版本 | 1.0 |
| 编 制 | 支付宝技术支持团队 |
| 负 责 人 | 支付宝技术支持团队 |
| 读 者 | 支付宝接口开发技术人员 |
| 保密级别 | 高 |

修订记录

| 版本号 | 修订日期 | 修订章节 | 修订人 | 修订原因 | 备注 |
|-----|-----------|------|--------|------|----|
| 1.0 | 2010.6.12 | 全部 | 技术支持团队 | 起草 | |

版权信息

本手册中所有的信息都是支付宝公司的财产并且受到严格的保密。未经过支付宝公司书面同意，接收本手册的人不能复制，公开，泄露手册的部分或全部的内容。

目录

| | |
|--|----|
| 修订记录..... | 2 |
| 目录 | 3 |
| 1 介绍..... | 4 |
| 1.1 类型..... | 4 |
| 1.2 适用人群..... | 4 |
| 1.3 文件说明..... | 4 |
| 2 准备工作..... | 5 |
| 2.1 确认工作..... | 5 |
| 2.1.1 确认商家网站的基本信息..... | 5 |
| 2.1.2 确认拿到的接口资料..... | 5 |
| 2.2 准备工作..... | 5 |
| 2.2.1 准备调试环境..... | 5 |
| 2.2.2 找到接入文件..... | 6 |
| 2.2.3 定位要传递的关键信息..... | 6 |
| 3 接入..... | 6 |
| 3.1 配置部分..... | 6 |
| 3.2 请求部分..... | 7 |
| 3.3 通知、返回部分..... | 8 |
| 4 调试..... | 8 |
| 4.1 步骤..... | 8 |
| 4.2 支付..... | 9 |
| 4.3 返回..... | 9 |
| 4.4 通知..... | 10 |
| 4.5 sign与mysign、veryfy_result的调试..... | 10 |
| 4.5.1 preg_match("/true\$/i",\$veryfy_result)不是true的情况 | 10 |
| 4.5.2 sign与mysign不等的情况 | 11 |
| 5 常见问题FAQ..... | 11 |

1 介绍

1.1 类型

接口类型: 通用即时到账接口

开发语言: php

编码格式: 适用 GBK (GB2312)、UTF-8 两种编码。

1.2 适用人群

本教程文档是面向了解 PHP 语言或其他语言等具有一定程序编写能力的技术人员的接口集成说明文档。

1.3 文件说明

文件包含以下几个文件。

1、index.php: (快速付款) 入口文件

用于在不涉及到购物车流程、充值流程等业务流程时, 只需要实现买家能够快速付款给卖家的付款功能。

2、alipayto.php: (确认订单) 入口文件

用来连接商户网站和支付宝收银台的主页面, 把商户网站的商品名称, 订单号, 以及其它的一些参数按照支付宝指定的规则传递给支付宝, 通过 form 表单 POST 或 URL GET 方式提交到支付宝。

3、alipay_config.php: 配置文件

用于设置支付宝账号信息, 配置文件路径等不需要动态变动的基本信息。配置文件将被接口中其他几个文件调用。

4、alipay_function: 公用函数文件

该文件为辅助请求处理文件(alipay_service.php)、通知返回处理文件(alipay_notify.php)的核心处理文件。它包含重复被调用的公共函数。不需要去修改。

5、alipay_service.php: 请求处理文件

该文件为辅助 index.php 文件的请求数据处理的的核心处理文件。不需要去修改。

6、alipay_notify.php: 通知返回处理文件

该文件为辅助返回文件、通知文件的通知返回数据处理的的核心处理文件。不需要去修改。

7、return_url.php: 同步返回处理文件

该文件的路径可以在 alipay_config.php 中通过 return_url 参数设定。付款完成以后, 支付宝会自动跳转到该页面。该页面用于处理付款完成以后支付宝返回的数据, 商家可以根据处理结果在商户网站上显示给用户或进入下一步流程操作。

8、notify_url.php: 异步通知处理文件

该文件的路径可以在 alipay_config.php 中通过 notify_url 参数设定。每次交易状态变更,

支付宝都会主动向该页面发出通知。该页面一般是在付款完成以后，返回文件没有执行数据处理的情况下，来对这笔交易做补救处理。

9、log.txt: 日志记录文件

用于记录对接口进行断点调试时的信息。可在接口程序的关键步骤处打开写日志函数，在程序执行后，便可根据在该文件中查到的程序执行情况来定位错误。

10、images: 支付宝图片文件夹

用于支付宝付款按钮的显示图片等图片。

2 准备工作

2.1 确认工作

2.1.1 确认商家网站的基本信息

- 确认合同是否包含即时到账接口功能。
合同查询，用签约的支付宝账号，登陆支付宝官方网站，点击“商家服务”→“合同列表”→“查看”。
- 确认网站的开发语言是否 PHP
建议询问购买网站的提供商、网站程序开发技术人员等。
- 确认网站编码格式是 GBK（GB2312）还是 UTF-8 的
可询问购买网站的提供商、网站程序开发技术人员等。
也可以通过在网站网页上右键，找到“编码”，打开扩展箭头后查看，默认选中的编码即为网站的编码格式。
- 确认网站的代码是否有被加密。
建议询问购买网站的提供商、网站程序开发技术人员等。

2.1.2 确认拿到的接口资料

- 确认拿到的是接口资料是即时到账的接口资料
- 确认拿到的接口资料是对应自己网站编码格式的 PHP 代码示例代码包。即自己的网站开发语言是 PHP、编码格式是 GBK（或 GB2312），那么拿到的接口资料也应该是开发语言是 PHP、编码格式是 GBK（或 GB2312）的；自己的网站开发语言是 PHP、编码格式是 UTF-8 的，那么拿到的接口资料也应该是开发语言是 PHP、编码格式是 UTF-8 的。

2.2 准备工作

2.2.1 准备调试环境

- 本机电脑的调试环境

开发工具: Zend Studio、NetBeans、Macromedia Dreamweaver 等;

语言环境: xampp 或 PHP 环境包;

编辑工具: EditPlus、NotePad++、电脑自带的“记事本”等工具

若使用的是开发软件,则只需要在开发软件中进行调试运行。

若使用的是 EditPlus、Macromedia Dreamweaver、记事本,则需要搭建 PHP 的环境,如: apache。

● 服务器的调试环境

1、服务器需要有支持 PHP 的环境

2、能够把本地文件上传到服务器上的通道与权限,如: FTP, SFTP 传输。

2.2.2 找到接入文件

一般网站只有在用户进行支付宝付款操作时才会调用支付宝接口,找到这个操作的按钮或链接(一般在网站的“确认订单”的页面上,如下图“立刻付款”按钮),将对应的调用页面设置为接口的 index.php。

| | |
|--|-------------------------|
| 订单编号: 20100521148078 查看详情 | 共需支付 ¥4245.00 |
| <hr/> | |
| 您选择了: 支付宝[即时到账] 选择其他支付方式» | 立刻付款 |

若是放在确认订单页面里,那么 Index.php 的页面就要与这个订单确认的页面进行融合,也就是把 index.php 的页面做成确认订单的页面。

2.2.3 定位要传递的关键信息

在商家网站的接入文件中,找到与订单有关的且需传递给支付宝的参数信息。

传递给支付宝的支付信息,除了接口里要求配置的基本信息外,还需要把与订单相关的数据: 商家网站订单号、总金额、订单名称(商品名称)、订单描述(商品描述)等在自己网站的订单系统中提取出来。

3 接入

3.1 配置部分

打开 alipay_config.php 文件,把所有信息全部填写完全。

- `$partner`、`$security_code` 需要用签约支付宝账号登陆支付宝官方网站,点击导航栏中“商家服务”后查到。

- `$notify_url`、`$return_url` 需要根据 `notify_url.php`、`return_url.php` 两个页面文件所在服务器的 `http://` 格式的完整路径来设定。
- `$show_url` 是订单商品展示页面，建议直接设置成商家网站的会员订单列表首页，如“我的订单”。若指向订单详细页面，会带上订单号的自定义参数（如：`show_url=http://www.xxx.com/alipay/order_detail.php?id=123`）有可能产生接口报错。
- `$input_charset`：商家网站可根据网站的编码（UTF-8 或 GBK/GB2312）下载对应编码的接口代码示例包。`$input_charset` 参数的值是不需要去修改的，若想修改建议直接下载另一个编码格式的接口代码示例包。
- 如果商家的服务器不支持 SSL，`$transport` 就选择 `http` 方式；反之，商家的服务器支持 SSL，`$transport` 可选择 `https`。
- `$antiphishing` 是防钓鱼功能开关，默认设置是关闭状态。若想使用该功能，首先服务器必须能够支持远程解析 XML 的功能（如：装有并配置启用 OpenSSL），其次综合考虑商家网站自身的实际情况是否有必要。该开关一旦打开，就不能再关上。即该功能一旦使用，以后使用即时到账时，都必须含有这个功能，否则接口使用会不正常。

3.2 请求部分

打开 `index.php` 文件，把“/*以下参数是需要通过下单时的订单数据传入进来获得*/”要求的部分全部设置完全。

在商家网站中，从下单的数据信息里提取出——订单号（`out_trade_no`）、订单名称（`subject`）、订单描述（`body`）、订单总金额（`total_fee`）、支付方式（`paymethod`）、网银选择（`defaultbank`）、自定义变量（`extra_common_param`）这些信息。以 POST 表单提交方式，在上一个下单流程页面中，传递到 `index.php` 文件里，那么这些数据都以 `$_POST['out_trade_no']`、`$_POST['subject']`、`$_POST['body']`、`$_POST['total_fee']`、`$_POST['paymethod']`、`$_POST['defaultbank']`、`$_POST['extra_common_param']`，这种形式得到传递过来的数据。即——

表单提交数据的页面代码：

```
<form id="alipaypay" name="alipaypay" action="../alipay/index.php" method="post"
target="_blank">
<input type="hidden" name="out_trade_no" value="<? php echo $out_trade_no; ?> "/>
<input type="hidden" name="subject" value="<? php echo $subject; ?> "/>
<input type="hidden" name="body" value="<? php echo $body; ?> "/>
<input type="hidden" name="total_fee" value="<? php echo $total_fee; ?> "/>
<input type="hidden" name="paymethod" value="<? php echo $paymethod; ?> "/>
<input type="hidden" name="defaultbank" value="<? php echo $defaultbank; ?> "/>
<input type="hidden" name="extra_common_param" value="<? php echo
$extra_common_param; ?> "/>
</form>
```

`index.php` 文件中获得数据的代码：

```
$out_trade_no = $_POST["out_trade_no"];
$subject      = $_POST["subject"];
$body         = $_POST["body"];
$total_fee    = $_POST["total_fee"];
```

```
$paymethod = $_POST["paymethod"];  
$defaultbank = $_POST["defaultbank"];  
$extra_common_param = $_POST["extra_common_param"];
```

3.3 通知、返回部分

打开 return_url.php 和 notify_url.php 文件，分别在 if(\$verify_result) “{}” 括号中加入商家网站自己的订单处理业务程序代码。

- 获取众多返回数据中最重要的数据，如：订单号 (out_trade_no)、订单名称 (subject)、订单描述 (body)、订单总金额 (total_fee)、交易状态 (trade_status)、支付宝交易号 (trade_no)、买家支付宝账号 (buyer_email)、自定义参数 (extra_common_param) 等
- 判断该笔订单是否有被处理过。由于同步通知和异步通知都会返回该笔订单的交易状态信息，那么该笔订单记录就有可能被重复刷新。因此，应先判断该笔订单记录中交易状态信息跟接收到的通知中的交易状态是否有做过处理，如果这笔订单没有做过处理则刷新该条记录；如果已经处理过该笔订单则不做处理。
- 编写业务订单业务流程处理程序代码。一般是在这里通过订单号，在订单数据库中更改该笔订单的相关数据，如订单状态从“未付款”改变成“已付款”等，并在订单数据库中存储此次订单对应的支付宝交易号是什么、买家账号是什么、处理时间是什么等。

注意：

- 1、由于能够根据订单号 (out_trade_no) SQL 语句查询得到商家订单数据库得到会员的相关信息或存储了会员的相关信息自定义参数 (extra_common_param)，所以不需要再在返回文件 return_url.php 做登陆验证等处理。
- 2、session、cookie 在返回文件中会失效，因此不要使用。
- 3、notify_url 文件里不要再做自动跳转，仅仅打印出 success 后，就不要再做任何程序执行。

4 调试

4.1 步骤

- 第一步：仅仅把接口配置好，不要放在网站项目中。在本机单独对这个接口进行调试。调试内容包括支付、返回；
- 第二步：本机调试没有问题后，再放入服务器中调试。调试内容包括：支付、返回、通知；
- 第三步：把调试好的接口与网站项目的下单流程进行衔接和融合；
- 第四步：在本机调试衔接到网站项目后的接口。调试内容包括：支付、返回；
- 第五步：本机调试没有问题后，再放入服务器中调试。调试内容包括：支付、返回、通知；

4.2 支付

- 第一步: 保证订单号 (out_trade_no) 的唯一性, 不要有重复的订单号出现。
- 第二步: 若包含下单流程, 须购买一件商品, 商品设定的金额 0.01 元, 下单完成。订单总额用 0.01 元做支付测试。
- 第三步: 访问 index.php 文件 (本机路径是: http://localhost/接口文件夹名/index.php; 服务器上路径是: http://网站域名/接口文件夹名/index.php) 即: 进入确认订单页面。



- 第四步: 点击支付宝付款的按钮, 页面会跳转至支付宝收银台;



- 第五步: 在支付宝的收银台里, 点击“确认无误, 付款”, 完成支付宝里的付款操作。一般情况下, 页面能成功到达支付宝收银台, 没有出错、没有乱码, 那么就不会有什么问题。

4.3 返回

- 第一步: 在支付宝收银台里, 完成付款操作, 支付宝页面显示付款成功, 如图:



- 第二步：等待几秒，页面自动跳转到返回页面（参数 `return_url` 对应的路径地址），查看返回的页面是否打印出返回的数据，根据返回的数据来判定是验证失败还是成功。

如果验证失败，

- 分步一：检查 `return_url` 的路径是不是对的
- 分步二：断点调试或用写日志函数记录代码执行情况
- 分步三：检查程序运行到 `alipay_notify.php` 文件的 `return_verify()` 函数中的 `if (preg_match("/true$/i",$verify_result) && $this->mysign == $_GET["sign"])` 这里时，是哪个判断出了问题。
- 第三步：断点调试或用写日志函数 `log_result()` 记录数据库程序代码段的执行情况。

4.4 通知

- 第一步：确定服务器是否开启写入权限。
- 第二步：服务器的 IP 地址是否有做过更换，详细可参考：
`http://club.alipay.com/read.php?tid=8687408`
- 第三步：必须把接口放置在服务器里做调试；
- 第四步：`notify_url` 的路径互联网上可以访问得到，且只显示 `fail`。其他如空白或是程序错误提示等内容，均属程序执行出错；
- 第五步：付款完成一次 1 分钱的交易；
- 第六步：用写日志函数 `log_result()` 记录 `alipay_notify.php` 文件的 `return_verify()` 函数中的 `if (preg_match("/true$/i",$verify_result) && $this->mysign == $_GET["sign"])`，这两个判断的执行结果
- 第七步：用写日志函数 `log_result()` 记录数据库程序代码段的执行情况；
- 第八步：用写日志函数 `log_result()` 记录整个页面打印出来的信息是否只有 `success`。

4.5 sign 与 mysign、verify_result 的调试

4.5.1 preg_match("/true\$/i",\$verify_result)不是 true 的情况

这种情况只与服务器设置问题、合作身份者 ID、`notify_id` 一分钟失效有关

- 1、合作身份者 ID（参数 `partner`）设置有错误

- 2、服务器有没有配置 SSL, 如果有, 那么 transport 参数应该等于 https; 若没有, 那么 transport 参数应该等于 http
- 3、DNS 解析, 是否能够解析支付宝 IP。支付宝 IP 范围是: 121.0.26.0~121.0.26.255 (支付宝主站所使用的 IP 地址是: 121.0.26.11, 支付宝通知信息的 IP 主地址: 121.0.26.1\121.0.26.2)
- 4、防火墙是否有拦截
端口 443 或 80 是否有开启, 且不是假死状态
- 5、返回的 url 有效时间是 1 分钟, 超过 1 分钟, 这个 URL 链接就会失效, \$verify_result 会为 false

4.5.2 sign 与 mysign 不等的情况

这种情况只与安全校验码、请求时的参数格式(如: 带自定义参数等)、编码格式有关。
调试工具: 写日志函数 `log_result`

- 1、安全校验码(参数 `security_code`) 设置有错误
- 2、请求参数 `show_url`、`return_url`、`notify_url` 是否有带自定义参数, 如:
`http://www.xxx.com/alipay/return_url.php?id=123`
`?id=123`, “?” 以后带的参数都是自定义参数
- 3、编码格式乱码。检查乱码的方法, 可用写日志函数来记录中文部分的数据是否乱码。
如果乱码则确定是编码格式设置不一致, 导致乱码。具体请检查编码格式参数 `_input_charset`、页面属性编码、网站项目整体编码等, 确保传递数据时没有乱码。

5 常见问题 FAQ

调试错误, 请回到请求来源地, 重新发起请求。

错误代码 `HAS_NO_PRIVILEGE`

解决方法: <http://club.alipay.com/read-htm-tid-8681436.html>

错误代码 `ILLEGAL_PARTNER`

解决方法: <http://club.alipay.com/read-htm-tid-8681433.html>

错误代码 `ILLEGAL_ARGUMENT`

解决方法: <http://club.alipay.com/read-htm-tid-8681434.html>

错误代码 `ILLEGAL_SIGN`

解决方法: <http://club.alipay.com/read-htm-tid-8681631.html>

错误代码 `seller_not_in_specified_sellers`

解决方法: <http://club.alipay.com/read-htm-tid-8681428.html>

错误代码 `ILLEGAL_LOGISTICS_FORMAT`

解决方法: <http://club.alipay.com/read-htm-tid-8681431.html>

错误代码 **TRADE_NOT_ALLOWED_PAY**

解决方法：<http://club.alipay.com/read-htm-tid-8681430.html>

错误代码 **SELLER_NOT_EXIST**

解决方法：<http://club.alipay.com/read-htm-tid-8681426.html>

错误代码 **BUYER_SELLER_EQUAL**

解决方法：<http://club.alipay.com/read-htm-tid-8681423.html>

- 更改服务器 IP 后，收不到 NOTIFY 通知问题

<http://club.alipay.com/read.php?tid=8687408>